Open IE as an Intermediate Structure for Semantic Tasks

Gabriel Stanovsky[†] **Ido Dagan**[‡] **Mausam**[§]
[†,‡]Department of Computer Science, Bar-Ilan University
[§]Department of Computer Science & Engg, Indian Institute of Technology, Delhi
[†]gabriel.satanovsky@gmail.com
[‡]dagan@cs.biu.ac.il
[§]mausam@cse.iitd.ac.in

Abstract

Semantic applications typically extract information from intermediate structures derived from sentences, such as dependency parse or semantic role labeling. In this paper, we study Open Information Extraction's (Open IE) output as an additional intermediate structure and find that for tasks such as text comprehension, word similarity and word analogy it can be very effective. Specifically, for word analogy, Open IE-based embeddings surpass the state of the art. We suggest that semantic applications will likely benefit from adding Open IE format to their set of potential sentence-level structures.

1 Introduction

Semantic applications, such as QA or summarization, typically extract sentence features from a derived intermediate structure. Common intermediate structures include: (1) Lexical representations, in which features are extracted from the original word sequence or the bag of words, (2) Stanford dependency parse trees (De Marneffe and Manning, 2008), which draw syntactic relations between words, and (3) Semantic role labeling (SRL), which extracts frames linking predicates with their semantic arguments (Carreras and Màrquez, 2005). For instance, a QA application can evaluate a question and a candidate answer by examining their lexical overlap (Pérez-Coutiño et al., 2006), by using short dependency paths as features to compare their syntactic relationships (Liang et al., 2013), or by using SRL to compare their predicate-argument structures (Shen and Lapata, 2007).

In a seemingly independent research direction, Open Information Extraction (Open IE) extracts coherent propositions from a sentence, each comprising a relation phrase and two or more argument phrases (Etzioni et al., 2008; Fader et al., 2011; Mausam et al., 2012). We observe that while Open IE is primarily used as an end goal in itself (e.g., (Fader et al., 2014)), it also makes certain structural design choices which differ from those made by dependency or SRL. For example, Open IE chooses different predicate and argument boundaries and assigns different relations between them.

Given the differences between Open IE and other intermediate structures (see Section 2), a research question arises: Can certain downstream applications gain additional benefits from utilizing Open IE structures? To answer this question we quantitatively evaluate the use of Open IE output against other dominant structures (Sections 3 and 4). For each of text comprehension, word similarity and word analogy tasks, we choose a state-of-the-art algorithm in which we can easily *swap* the intermediate structure while preserving the algorithmic computations over the features extracted from it. We find that in several tasks Open IE substantially outperforms other structures, suggesting that it can provide an additional set of useful sentence-level features.

2 Intermediate Structures

In this section we review how intermediate structures differ from each other, in terms of their imposed structure, predicate and argument boundaries, and the type of relations that they introduce. We include Open IE in this analysis, along with lexical, dependency and SRL representations, and highlight its unique properties. As we show in Section 4, these differences have an impact on the overall performance of certain downstream applications.

Lexical representations introduce little or no structure over the input text. Features for following computations are extracted directly from the original word sequence, e.g., word count statistics or lexical overlap (see Figure 1a).

Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), pages 303–308, Beijing, China, July 26-31, 2015.

Syntactic dependencies impose a tree structure (see Figure 1b), and use words as atomic elements. This structure implies that predicates are generally composed of a single word and that arguments are computed either as single words or as entire spans of subtrees subordinate to the predicate word.

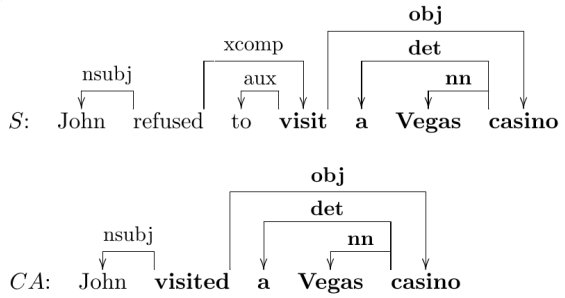
In SRL (see Figure 1c), several non-connected frames are extracted from the sentence. The atomic elements of each frame consist of a single-word predicate (e.g., the different frames for *visit* and *refused*), and a list of its semantic arguments, without marking their internal structure. Each argument is listed along with its semantic relation (e.g., *agent*, *instrument*, etc.) and usually spans several words.

Open IE (see Figure 1d) also extracts non-connected propositions, consisting of a predicate and its arguments. In contrast to SRL, argument relations are not analyzed, and predicates (as well as arguments) may consist of several consecutive words. Since Open IE focuses on human-readability, infinitive constructions (e.g., *refused to visit*), and multi-word predicates (e.g., *took advantage*) are grouped in a single predicate slot. Additionally, arguments are truncated in cases such as prepositional phrases and reduced relative clauses. The resulting structure can be understood as an extension of shallow syntactic chunking (Abney, 1992), where chunks are labeled as either predicates or arguments, and are then interlinked to form a complete proposition.

It is not clear apriory whether the differences manifested in Open IE's structure could be beneficial as intermediate structures for downstream applications. Although a few end tasks have made use of Open IE's output (Christensen et al., 2013; Balasubramanian et al., 2013), there has been no systematic comparison against other structures. In the following sections, we quantitatively study and analyze the value of Open IE structures against the more common intermediate structures – lexical, dependency and SRL, for three downstream NLP tasks.

S**:** John refused to **visit a Vegas casino**
CA**:** John **visited a Vegas casino**

(a) Lexical matching of a 5 words window (marked with a box). Current window yields a score of 4 - words contributing to the score are marked in bold.



(b) Dependency matching yields a score of 3. Contributing triplets are marked in bold.

S**:** refused$_{0.1}$: **A$_0$: John** A$_1$: to visit a Vegas casino
visit$_{0.1}$: A$_0$: John A$_1$: a Vegas casino
CA**: visit$_{0.1}$: A$_0$: John A$_1$: a Vegas casino**

(c) SRL frames matching yields a score of 4, frame elements contributing to the score marked in bold.

S**:** (**John**, refused to visit, **a Vegas casino**)
CA**:** (**John**, visited, **a Vegas casino**)

(d) Open IE matching yields a score of 2, contributing entries marked in bold.

Figure 1: Different intermediate structures used to compute the modified text comprehension matching score (Section 3), when answering a question *"Where did John visit?"*, given an input sentence S: *"John refused to visit a Vegas casino"*, and a *wrong* candidate answer CA: *"John visited a Vegas casino"*.

3 Tasks and Algorithms

Comparing the effectiveness of intermediate structures in semantic applications is hard for several reasons: (1) extracting the underlying structure depends on the accuracy of the specific system used, (2) the overall performance in the task depends heavily on the computations carried on top of these structures, and (3) different structures may be suitable for different tasks. To mitigate these complications, and comparatively evaluate the effectiveness of different types of structures, we choose three semantic tasks along with state-of-the-art algorithms that make a clear separation between feature extraction and subsequent computation. We then compare performance by using features from four intermediate structures – lexical, dependency, SRL and Open IE. Each of these is extracted using state-of-the-art systems. Thus, while our comparisons are valid only for the tested tasks and systems, they do provide valuable evidence for the general question of effective intermediate structures.

3.1 Text Comprehension Task

Text comprehension tasks extrinsically test natural language understanding through question answer-

Target	Lexical	Dependency	SRL	Open IE
	John	nsubj_John	A0_John	0_John
	to	xcomp_visit	A1_to	1_to
refused	visit		A1_visit	1_visit
	Vegas		A1_Vegas	2_Vegas

Table 1: Some of the different contexts for the target word "refused" in the sentence *"John refused to visit Vegas"*. SRL and Open IE contexts are preceded by their element (predicate or argument) index. See figure 1 for the different representations of this sentence.

ing. We use the MCTest corpus (Richardson et al., 2013), which is composed of short stories followed by multiple choice questions. The MCTest task does not require extensive world knowledge, which makes it ideal for testing underlying sentence representations, as performance will mostly depend on accuracy and informativeness of the extracted structures.

We adapt the unsupervised lexical matching algorithm from the original MCTest paper. It counts lexical matches between an assertion obtained from a candidate answer (CA) and a sliding window over the story. The selected answer is the one for which the maximum number of matches are found. Our adaptation changes the algorithm to compute a modified *matching score* by counting matches between *structure units*. The corresponding units are either dependency edges, SRL frame elements or Open IE tuple elements. Figure 1 illustrates computations for a sentence - candidate answer pair.

3.2 Similarity and Analogy Tasks

Word similarity tasks deal with assessing the degree of "similarity" between two input words. Turney (2012) classifies two types of similarity: (1) domain similarity, e.g., *carpenter* is similar to *wood, hammer*, and *nail*, (2) functional similarity, in which *carpenter* will be similar to other professions, e.g., *shoemaker, brewer, miner* etc. Several evaluation test sets exist for this task, each targeting a slightly different aspect of similarity. While Bruni (2012), Luong (2013), Radinsky (2011), and ws353 (Finkelstein et al., 2001) can be largely categorized as targeting domain similarity, simlex999 (Hill et al., 2014) specifically targets functional aspects of similarity (e.g., *coast* will be similar to *shore*, while *closet* will not be similar to *clothes*). A related task is *word analogy*, in which systems take three input words (A:A^*, B:?) and output a word B^*, such that the relation between B and B^* is closest to the relation between A and A^*. For instance, *queen* is the desired answer for the triple (*man*:*king*, *woman*:?).

Some recent state-of-the-art approaches to these two tasks derive a similarity score via arithmetic computations on word embeddings (Mikolov et al., 2013b). While original training of word embeddings used lexical contexts (n-grams), recently Levy and Goldberg (2014) generalized this to arbitrary contexts, such as dependency paths. We use their software[1] and recompute the word embeddings using contexts from our four structures: lexical context, dependency paths, SRL's semantic relations, and Open IE's surrounding tuple elements. Table 1 shows the different contexts for a sample word.

4 Evaluation

In our experiments we use MaltParser (Nivre et al., 2007) for dependency parsing, and ClearNLP (Choi and Palmer, 2011) for SRL.

To obtain Open-IE structures, we use the recent Open IE-4 system[2] which produces n-ary extractions of both verb-based relation phrases using SRLIE (an improvement over (Christensen et al., 2011)) and nominal relations using regular expressions. SRLIE first processes sentences using SRL and then uses hand-coded rules to convert SRL frames and associated dependency parses to open extractions.

We choose these tools as they are on par with state-of-the-art in their respective fields, and therefore represent the current available off-the-shelf intermediate structures for semantic applications. Furthermore, Open IE-4 is based on ClearNLP's SRL, allowing for a direct comparison. For SRL systems, we take argument boundaries as their complete parse subtrees.[3]

Results on Text Comprehension Task We report results (in percentage of correct answers) on the whole of MC500 dataset (ignoring train-dev-test split) since all our methods are unsupervised. Figure 2 shows the accuracies obtained on the multiple-choice questions, categorized by *single* (the question can be answered based on a sin-

[1] *https://bitbucket.org/yoavgo/word2vecf*

[2] *http://knowitall.github.io/openie/*

[3] We tried an alternative approach which takes only the heads as arguments, but that performed much worse.

	Open IE	Lexical	Deps	SRL
bruni	**.757**	.735	.618	.491
luong	**.288**	.229	.197	.171
radinsky	**.681**	.674	.592	.433
simlex	.39	.365	**.447**	.306
ws353-rel	**.647**	.64	.492	.551
ws353-sym	**.77**	.763	.759	.439
ws353-full	**.711**	.703	.629	.693

Table 2: Performance in word similarity tasks (Spearman's ρ)

	Google		MSR	
	Add	Mul	Add	Mul
Open IE	**.714**	**.719**	**.529**	**.55**
Lexical	.651	.656	.438	.455
Deps	.34	.367	.4	.434
SRL	.352	.362	.389	.406

Table 3: Performance in word analogy tasks (percentage of correct answers)

gle story sentence) , *multiple* (multiple sentences needed) and *all* (*single* + *multiple*).[4]

In this task, we find that Open IE and dependency edges substantially outperform lexical and SRL. We conjecture that SRL's weak performance is due to its treatment of infinitives and multi-word predicates as different propositions (see Section 2). This adds noise by wrongly counting partial matching between predications, as exemplified in Figure 1c. The gain over the lexical approach can be explained by the ability to capture longer range relations than the fixed size window.[5] In our results Open IE slightly improves over dependency. This can be traced back to the different structural choices depicted in Section 2 – Open IE counts matches at the proposition level while the dependency variant may count path matches over unrelated sentence parts. The differences between the performance of Open IE and all other systems were found to be statistically significant ($p < 0.01$).

Results on Similarity and Analogy Tasks For these tasks, we train the various word embeddings on a Wikipedia dump (August 2013 dump), containing 77.5M sentences and 1.5B tokens. We used the default hyperparameters from Levy and Goldberg (2014): 300 dimensions, skip gram with negative sampling of size 5. Lexical embeddings were trained with 5-gram contexts. Performance is measured using Spearman's ρ, in order to assess the correlation of the predictions to the gold annotations, rather than comparing their values directly. Table 2 compares the results on the *word similarity task* using cosine similarity between embeddings as the similarity predictor. For the *ws353* test set we report results on the whole corpus (*full*) as well as on the partition suggested by (Agirre et al., 2009) into *relatedness* (mainly meronym-holonym) and *similarity* (synonyms, antonyms, or hyponym-hypernym).

We find that Open IE-based embeddings consistently do well; performing best across all test sets, except for simlex999. Analysis reveals that Open IE's ability to represent multi-word predicates and arguments allows it to naturally incorporate *both* notions of similarity. Context words originating from the same Open IE slot (either predicate or argument) are lexically close and indicate domain-similarity, whereas context words from other elements in the tuple express semantic relationships, and target functional similarity.

Thus, Open IE performs better on word-pairs which exhibit both topical and functional similarity, such as *(latinist, classicist)*, or *(provincialism, narrow-mindedness)*, which were taken from the Luong test set. Table 4 further illustrates this dual capturing of both types of similarity in Open IE space.

Our results also reiterate previous findings – lexical contexts do well on domain-similarity test sets (Mikolov et al., 2013b). The results on the simlex999 test set can be explained by its focus on functional similarity, previously identified as better captured by dependency contexts (Levy and Goldberg, 2014).

For the *Word analogy task* we use the Google (Mikolov et al., 2013a) and the Microsoft corpora (Mikolov et al., 2013b), which are composed of $\sim 195K$ and $8K$ instances respectively. We obtain the analogy vectors using both the additive and multiplicative measures (Mikolov et al., 2013b; Levy and Goldberg, 2014). Table 3 shows the results – Open IE obtains the best accuracies by vast margins ($p < 0.01$), for reasons simi-

[4] As expected, all sentence-level intermediate structures perform best on the *single* partition, yet results show that some of the questions from the *multiple* partition may also be answered correctly using information from a single sentence.

[5] We experimented with various window sizes and found that window size of the length of the current candidate-answer performed best.

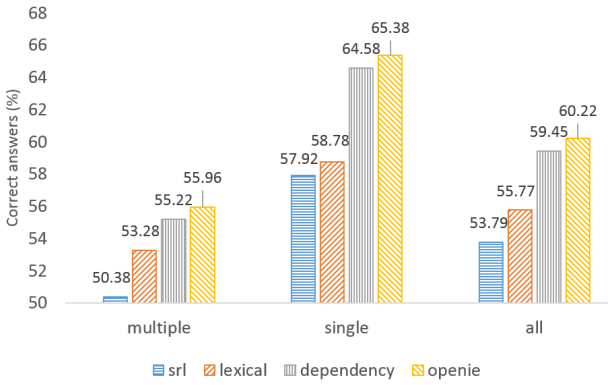


Figure 2: Performance in MCTest (percentage of correct answers).

lar to the word similarity tasks. To our knowledge, Open IE results on both analogy datasets surpass the state of the art. An example (from the Microsoft test set) which supports the observation regarding Open IE embeddings space is (*gentlest*:*gentler*, *loudest*:?), for which only Open IE answers correctly as *louder*, while lexical respond with *higher-pitched* (domain similar to *loudest*), and dependency with *thinnest* (functionally similar to *loudest*). Our Open-IE embeddings are freely available[6] and we note that these can serve as plug-in features for other NLP applications, as demonstrated in (Turian et al., 2010).

5 Conclusions

We studied Open IE's output compared with other dominant structures, highlighting their main differences. We then conduct experiments and analysis suggesting that these structural differences prove beneficial for certain downstream semantic applications. A key strength is Open IE's ability to balance lexical proximity with long range dependencies in a single representation. Specifically, for the word analogy task, Open IE-based embeddings surpass all prior results. We conclude that an NLP practitioner will likely benefit from adding Open IE to their toolkit of potential sentence representations.

[6] *http://www.cs.bgu.ac.il/~gabriels*

Target Word	Lexical	Dependency	Open IE
canine	dog	feline	dog
	incisor	bovine	carnassial
	dentition	equine	feline
	parvovirus	porcine	fang-like
	dysplasia	murine	bovine

Table 4: Closest words to *canine* in various word embeddings. Illustrating domain similarity (Lexical), functional similarity (Dependency), and a mixture of both (Open IE).

Acknowledgments

This work was partially supported by the Israel Science Foundation grant 880/12, the German Research Foundation through the German-Israeli Project Cooperation (DIP, grant DA 1600/1-1), a Google Research Award to Ido Dagan, and Google's Language Understanding and Knowledge Discovery Focused Research Award to Mausam.

References

Steven P Abney. 1992. Parsing by chunks. *Principle-based parsing*, pages 257–278.

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27. Association for Computational Linguistics.

Niranjan Balasubramanian, Stephen Soderland, Mausam, and Oren Etzioni. 2013. Generating coherent event schemas at scale. In *Conference on Empirical Methods in Natural Language Processing*, pages 1721–1731.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 136–145. Association for Computational Linguistics.

Xavier Carreras and Lluís Màrquez. 2005. Introduction to the conll-2005 shared task: Semantic role labeling. In *Proceedings of The SIGNLL Conference on Computational Natural Language Learning (CoNLL)*, pages 152–164.

Jinho D Choi and Martha Palmer. 2011. Transition-based semantic role labeling using predicate argument clustering. In *Proceedings of the ACL 2011 Workshop on Relational Models of Semantics*, pages 37–45. Association for Computational Linguistics.

Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2011. An analysis of open information extraction based on semantic role labeling. In *Proceedings of the 6th International Conference on Knowledge Capture (K-CAP '11)*.

Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2013. Towards coherent multi-document summarization. In *Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies*, pages 1163–1173.

Marie-Catherine De Marneffe and Christopher D Manning. 2008. The stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S Weld. 2008. Open information extraction from the Web. *Communications of the ACM*, 51(12):68–74.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014. Open question answering over curated and extracted knowledge bases. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 1156–1165.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

Felix Hill, Roi Reichart, and Anna Korhonen. 2014. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *arXiv preprint arXiv:1408.3456*.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2.

Percy Liang, Michael I. Jordan, and Dan Klein. 2013. Learning dependency-based compositional semantics. *Computational Linguistics*, 39(2):389–446.

Minh-Thang Luong, Richard Socher, and Christopher D. Manning. 2013. Better word representations with recursive neural networks for morphology. *The SIGNLL Conference on Computational Natural Language Learning (CoNLL)*, 104.

Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534, Jeju Island, Korea, July. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Workshop at The International Conference on Learning Representations (ICLR)*.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies*, pages 746–751.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(02):95–135.

Manuel Pérez-Coutiño, Manuel Montes-y Gómez, Aurelio López-López, and Luis Villaseñor-Pineda. 2006. *The role of lexical features in Question Answering for Spanish*. Springer.

Kira Radinsky, Eugene Agichtein, Evgeniy Gabrilovich, and Shaul Markovitch. 2011. A word at a time: computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th international conference on World wide web*, pages 337–346. ACM.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Conference on Empirical Methods in Natural Language Processing*, pages 193–203.

Dan Shen and Mirella Lapata. 2007. Using semantic roles to improve question answering. In *EMNLP-CoNLL 2007, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June 28-30, 2007, Prague, Czech Republic*, pages 12–21.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics.

Peter D Turney. 2012. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research*, 44:533–585.